

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended): A software package verification tool for verifying a software installation package that includes at least one software component without performing a test execution of a software component of the software package wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the tool comprising:

at least one test module ~~configured to use the data entries of the file list to test at least one parameter of the software package;~~

a framework operable to identify the at least one test module defining a test of at least one parameter of the at least one software component of the package, wherein the test modules do not require the test execution of software components of the software installation package;

and

a control module operable to access the framework to cause the at least one test module identified therein to perform the test defined thereby for verifying the package.

Claim 2 (original): The tool of claim 1, wherein the framework identifies a plurality of test modules.

Claim 3 (original): The tool of claim 2, wherein the framework identifies a priority for each of the test modules.

Claim 4 (original): The tool of claim 3, wherein the control module is operable to cause the test modules to be executed sequentially according to the priority identified in the framework for each of the test modules.

Claim 5 (original): The tool of claim 1, wherein a mechanism is provided for identifying the at least one test module as being one of active and not active.

Claim 6 (original): The tool of claim 5, wherein the mechanism for identifying the at least one test module as being one of active and not active is included in the framework.

Claim 7 (original): The tool of claim 5, wherein the mechanism for identifying the at least one test module as being one of active and not active is included in the control module.

Claim 8 (original): The tool of claim 2, wherein the framework comprises a directory having a plurality of entries, each entry identifying one of the plurality of test modules.

Claim 9 (original): The tool of claim 8, wherein entry defines a priority for the one of the test modules identified therein.

Claim 10 (original). The tool of claim 8, wherein the identity of the one of the test modules defines its priority.

Claim 11 (original): The tool of claim 2, wherein each of the plurality of test modules is formed by a script and the framework identifies each of the test modules by a name for the script.

Claim 12 (original): The tool of claim 2, wherein each of the plurality of test modules is formed by a software object.

Claim 13 (currently amended): A computer program on a carrier medium for verifying a software package that includes at least one software component without performing a test execution of a software component of the software package ~~wherein the software package includes a file list having data entries associated with parameters for the at least one software component~~, the computer program comprising computer executable instructions for:

~~a) providing at least one test module configured to use the data entries of the file list to test at least one parameter of the software package;]]~~

b) a) forming a framework operable to identify at least one test module for testing defining a test of at least one parameter of the at least one software component of the package, wherein the test module does not require the test execution of a software component of the software package;

and

b) e) forming a control module operable to access the framework to cause the at least one test module identified therein to perform the test for verifying the package.

Claim 14 (currently amended): A program storage device readable by a computer, tangibly embodying a program of instructions executable by the computer to perform method steps for verifying a software installation package without performing a test execution of a software component of the software installation package that includes at least one software component and wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the method comprising the steps of:

a) providing a framework for identifying at least one test module, each said test module defining a test of at least one parameter of the at least one software component of the package wherein said test is configured to use the data entries of the file list to test the at least one parameter of the software package and wherein the testing is accomplished without performing a test execution of a software component of the software installation package;

b) accessing the framework to identify the at least one test module: and

c) causing the at least one test module to perform the test defined thereby on the package.

Claim 15 (original): The method of claim 14, wherein the framework identifies a plurality of test modules.

Claim 16 (original): The method of claim 15, wherein a priority for each of the test modules is identified in the framework.

Claim 17 (original): The method of claim 15, comprising sequentially causing each of the test modules to be executed according to the priority identified for each of the test modules.

Claim 18 (original): The method of claim 15, comprising identifying each of the test modules as being one of active and not active.

Claim 19 (original): The method of claim 15, comprising providing a directory in the framework, wherein the directory has a plurality of entries, each entry identifying one of the plurality of test modules.

Claim 20 (currently amended): A system for verifying a software installation package without performing a test execution of a software component of the software installation package, the system including that includes at least one software component wherein the software installation package includes a file list having data entries associated with parameters for the at least one software component, the system comprising:

- a) at least one test module configured to use the data entries of the file list to test at least one parameter of the software installation package without performing a test execution of a software component of the software installation package;
- b) a framework to identify at least one test module;
- and
- c) a control module operable to access the framework for causing the at least one test module identified therein to perform the test defined thereby for verifying the package.

Claim 21 (original): The system of claim 20, wherein the system comprises a computer including a processor, memory and software held in memory and operable to control the processor, the software forming:

said framework and said control module.

Claim 22 (currently amended): A computer system for verifying a software installation package without performing a test execution of software components of the software installation package that includes at least one software component wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the system comprising:

- a) a memory for storing software; and
- b) a processing unit for executing the software to carry out the steps of:
 - (i) providing a framework to identify at least one test module defining a test of at least one parameter of the at least one software component of the package;
 - and
 - (ii) providing a control module operable to access the framework for causing the at least one test module identified therein to perform a test that uses the data entries of the file list to test the at least one parameter of the software package thereby verifying the package wherein the file list data entries are associated with parameters concerning at least one of: compiler versions used with the software components, copyright information

concerning the software components, the size of the software components, the binary data types of the software components.

Claim 23 (currently amended): A method for verifying a software installation package that includes at least one software component wherein verifying is conducted without performing a test execution of the software components of the software installation package and wherein the software package includes a file list having data entries associated with parameters for the at least one software component, the method comprising the steps of:

a) providing a framework for identifying at least one test module, each said test module configured to use the data entries of the file list to test at least one parameter of the software package thereby defining a test of at least one parameter of the at least one software component of the package;

b) accessing the framework to identify the at least one test module; and

c) causing the at least one test module to perform the test defined thereby on the package.

Claim 24 (original): The method of claim 23, wherein the framework identifies a plurality of test modules.

Claim 25 (original): The method of Claim 24, wherein a priority for each of the test modules is identified in the framework.

Claim 26 (original): The method of claim 25, comprising sequentially causing each of the test modules to be executed according to the priority identified for each of the test modules.

Claim 27 (original): The method of claim 24, comprising identifying each of the test modules as being one of active and not active.

Claim 28 (original): The method of claim 24, comprising providing a directory in the framework, wherein the directory has a plurality of entries, each entry identifying one of the test modules.

Claim 29 (original): The method of claim 28, wherein each entry defines a priority of the test module identified thereby.

Claim 30 (original): The method of claim 28, wherein identity of a module defines its priority.

Claim 31 (currently amended): A method of verifying a software package that includes at least one software component, the method comprising the steps of:

- a) receiving the software package wherein the software package includes a file list having data entries associated with parameters for the at least one software component;
- b) accessing a framework that references at least one test module to identify the at least one test module from the framework, each said test module configured to use the data entries of the file list to define a test of the software package; and
- c) performing the test defined by the at least one test module on the package wherein testing is conducted without performing a test execution of the software components of the software installation package.

Claim 32 (previously amended): The method of Claim 31, including repeating steps (b) and (c) to perform a sequence of tests, the order in which the tests are performed being determined by relative priorities assigned to each of the at least one test module.

Claim 33 (original): A computer readable medium having stored thereon a data structure operable for us in verifying a software package that includes at least one software component, the data structure comprising:

- a) a first field containing data representing one of a plurality of test modules, each test module being operable to test of at least one parameter of the at least one software component of the package,
- b) where data representing ones of the test modules may be added to and deleted from the data structure, creating a flexible data structure.

Claim 34 (original): The medium according to claim 33, wherein the data structure further comprises a second field identifying a priority for each of the test modules represented by the data in the first data field, the priority defining an order of execution of test modules.

Claim 35 (original): The medium according to claim 33, wherein the data structure further comprises a third field identifying the one of a plurality of test modules represented by the data in the first data field as being one of active and not active.

Claim 36 (previously presented): A software package verification tool as in Claim 1 wherein the software package is compliant with the SOLARIS standard.

Claim 37 (previously presented): A software package verification tool as in Claim 36 wherein the file list comprises a "pkgmap" file.

Claim 38 (new): A method of verifying a that software installation package that includes at least one software component, the method comprising the steps of:

- a) providing a test framework that includes a control module and at least one test module suitable for conducting verification of software installation packages;
- b) receiving the software installation package;
- c) executing an initial verification of the software installation package using a control module, wherein the initial verification is conducted without performing a test execution of the software components of the software installation package;
- d) where the initial verification is successful, selecting a next module from among the at least one test module to conduct verification of the software package; and
- e) executing verification testing of the software package using the next module wherein the verification using the next module is conducted without performing a test execution of the software components of the software installation package.

Claim 39 (new): The method of Claim 38 wherein operations d and e are iteratively performed until all test modules of the at least one test module have been executed.

Claim 40 (new): The method of Claim 38 wherein if any verification operations fail an error message is generated.

Claim 41 (new): The method of Claim 38 wherein executing the initial verification of the software installation package comprises:

- receiving a verification command; and
- checking that the verification command includes a correct number of arguments.

Claim 42 (new): The method of Claim 41 wherein executing the initial verification of the software installation package comprises at least one of:

- confirming that a user of the software installation package has the correct permissions;
- confirming that software components of the software installation package exist in the correct directories;

- confirming that the software components do not include any zero size files;

- confirming that the software components comprise actual data files and not data links;

and

- confirming that package map and package information files for the software package exist in a Solaris environment.

Claim 43 (new): The method of Claim 38 wherein a) providing a test framework that includes a control module and at least one test module suitable for conducting verification of software installation packages includes adding, deleting, and modifying the at least one test module to provide a flexible test framework.

Claim 44 (new): The tool of claim 2, wherein the plurality of test modules of the framework define a series of automatically executable tests and wherein the test modules do not require the test execution of software components of the software installation package.

Claim 45 (new): The tool of claim 44, wherein the software installation package comprises a file list that identifies a plurality of software components included in the software package and includes data about the software components of the software package; and wherein framework identifies a plurality of test modules.

Claim 46 (new): The tool of claim 44, wherein the file list that identifies the plurality of software components in the software package comprises a Solaris pkgmap file.

Claim 47 (new): The tool of claim 44, wherein the plurality of test modules are operable to use the data included in the file list to verify the software package.

Claim 48 (new): The tool of claim 47, wherein the data included in the file list includes parameter information concerning the plurality of software components in the software installation package; and

wherein the plurality of test modules are operable to use the parameter information included in the file list to verify the software package.

Claim 49 (new): The tool of claim 48, wherein the parameter information in the file list includes at least one of: file names for the software components, version numbers for the software components, vendor identification for the software components, copyright information concerning the software components, the size of the software components, the binary data types of the software components.

Claim 50 (new): The tool of claim 48, wherein the parameter information in the file list includes the compiler version used with the software components.

Claim 51 (new): The tool of claim 48, wherein the parameter information in the file list includes at least one of: copyright information concerning the software installation components, the size of the software components, the binary data types of the software components.

Claim 52 (new): The tool of Claim 2 wherein test modules of the framework can be added, deleted, or modified creating a flexible framework.

Claim 53 (new): The tool of claim 2, wherein test modules can be added and deleted from the framework and can be modified to provide a flexible framework.

Claim 54 (new): The method of claim 15, wherein the file list that identifies the plurality of software components in the software package comprises a Solaris pkgmap file.

Claim 55 (new): The method of claim 54, wherein c) causing the at least one test module to perform the test defined thereby on the package includes performing the test using parameters included in the file list to verify the software package.

Claim 56 (new): The method of claim 55, wherein the parameter information in the file list includes at least one of: file names for the software components, version numbers for the

software components, vendor identification for the software components, copyright information concerning the software components, the size of the software components, the binary data types of the software components.

Claim 57 (new): The method of claim 55, wherein the parameter information in the file list includes the compiler version used with the software components.

Claim 58 (new): The method of claim 55, wherein the parameter information in the file list includes at least one of: copyright information concerning the software installation components, the size of the software components, the binary data types of the software components.

Claim 59 (new): The method of claim 55, wherein the providing the framework includes added and deleted test modules from the framework as desired.

Claim 60 (new): The system of claim 21, wherein the test modules are configured to verify the software installation package by testing the software components of the software installation package using data entries of the file list to test parameters including at least one of: file names for the software components, version numbers for the software components, vendor identification for the software components, copyright information concerning the software components, the size of the software components, the binary data types of the software components.

Claim 61 (new): The system of claim 21, wherein the test modules are configured to verify the software installation package by testing data entries of the file list to determine if the correct compiler version is used with the software components.

Claim 62 (new): The system of claim 21, wherein the test modules are configured to verify the software installation package by testing data entries of the file list to determine if the parameter information in the file list includes correct information concerning at least one of: copyright information concerning the software installation components, the size of the software components, the binary data types of the software components.

Claim 63 (new): The method of claim 23, wherein the plurality of test modules are configured to test at least one of: that the software components are compiled using the same

compiler version; that binary data types of the software components are compatible with the same architecture; that the copyrights are current; and a test whether only specified software components are present in the installation package.

Claim 64 (new): The method of claim 63, wherein the plurality of test modules are further configured to include a module for testing whether there have been changes to the software installation package relative to a prior version of the software installation package.

Claim 65 (new): The method of claim 63, wherein the plurality of test modules are further configured to include a module for testing whether there are any zero size files in the software installation package.